

- 1 1. A method of graphically rendering a virtual object, the method comprising the  
2 steps of:
  - 3 (a) using an index corresponding to each of a plurality of jacks of a voxel-  
4 based virtual object to identify texture elements for which surface elements of the virtual  
5 object are mapped; and
  - 6 (b) generating texture coordinates of the identified texture elements in a  
7 texture space.
- 1 2. The method of claim 1, wherein the index corresponds to one of a known plurality  
2 of surface element configurations produced during mesh generation.
- 1 3. The method of claim 1, wherein the index is a marching cubes index or a  
2 marching tetrahedra index.
- 1 4. The method of claim 1, wherein the texture space comprises a plurality of texture  
2 regions.
- 1 5. The method of claim 4, further comprising the steps of:
  - 2 (c) binding to a graphics application a blended texture corresponding to one  
3 of the plurality of texture regions; and
  - 4 (d) transmitting to the graphics application the texture coordinates of the  
5 texture elements of the blended texture.
- 1 6. The method of claim 1, further comprising the step of:
  - 2 (c) dividing an object space comprising at least a portion of the virtual object  
3 into a plurality of jack blocks each comprising a plurality of jacks, wherein step (b)

4 comprises generating texture coordinates of texture elements to which surface elements  
5 within each jack of each jack block are mapped.

1 7. The method of claim 1, wherein step (a) comprises using a first lookup table to  
2 determine to which of a plurality of texture elements a surface element is mapped.

1 8. The method of claim 7, wherein step (a) comprises using a second lookup table to  
2 determine to which of a plurality of positions within a texture element a surface element  
3 is mapped.

1 9. The method of claim 7, further comprising the step of creating the first lookup  
2 table.

1 10. The method of claim 1, wherein the texture elements are quadrilaterals and the  
2 surface elements are triangles.

1 11. The method of claim 10, wherein step (a) comprises assigning either one or two  
2 triangular surface elements to a quadrilateral texture element.

1 12. The method of claim 11, wherein step (a) comprises assigning to a quadrilateral  
2 texture element two triangular surface elements that share a common edge.

1 13. The method of claim 1, wherein the surface elements of the virtual object are  
2 components of a triangle mesh generated for the virtual object using at least one of a  
3 marching cubes algorithm and a marching tetrahedra algorithm. ✓

1 14. A method of mapping texture onto a virtual object, the method comprising the  
2 steps of:

3 (a) at least one of:

4 (i) allocating texture in a texture space for at least one newly-created  
5 jack of a virtual object following an object modification; and

6                   (ii)     de-allocating texture in the texture space for at least one newly-  
7                   eliminated jack of the virtual object following an object modification; and

8                   (b)     rendering the virtual object.

1    15.     The method of claim 14, wherein the texture is spatially incoherent.

1    16.     The method of claim 14, wherein step (a) comprises determining, in a stepwise  
2    manner for each of a plurality of jacks, whether texture elements have previously been  
3    assigned to surface elements in that jack.

1    17.     The method of claim 16, wherein step (a) proceeds from a first jack to a second  
2    jack in render order.

1    18.     The method of claim 16, wherein step (a) comprises determining whether there  
2    are sufficient free texture elements in a current texture region in which to allocate texture  
3    for the jack.

1    19.     The method of claim 18, wherein step (a) comprises creating a new texture region  
2    where there are insufficient free texture elements in the current texture region for the  
3    jack.

1    20.     The method of claim 19, wherein step (a) comprises keeping within a single  
2    texture region all texture elements assigned to all jacks of a jack block.

1    21.     The method of claim 14, wherein step (b) comprises generating, for surface  
2    elements of the virtual object, coordinates of corresponding texture elements in the  
3    texture space determined according to a marching cubes index for each of a plurality of  
4    jacks of the virtual object.

1    22.     The method of claim 14, wherein step (a) and step (b) are each performed  
2    automatically, and wherein the method obviates creating a global parameterization.

1 23. The method of claim 14, wherein step (a) comprises retaining texture previously  
2 allocated for at least one jack that is unchanged following the object modification.

1 24. The method of claim 14, wherein step (a) comprises generating, for a given  
2 texture region, a hash map comprising an identification of each of at least one texture  
3 element in each of a plurality of jacks.

1 25. The method of claim 14, wherein step (a) comprises generating, for a given  
2 texture region, a hash map of hash maps comprising an identification of each of at least  
3 one texture element in each jack of at least one jack block.

1 26. A method of creating a blended texture for use in rendering a virtual object, the  
2 method comprising the step of blending a plurality of texture layers, at least one of which  
3 has associated therewith a grid indicating at least one member of each of at least two of:

4 (i) a uniform texture element;

5 (ii) a nonuniform texture element; and

6 (iii) a location of a nearest free texture element in the grid.

1 27. The method of claim 26, wherein the grid indicates at least one member of each of  
2 (i), (ii), and (iii).

1 28. The method of claim 26, wherein an element of the grid indicating a uniform  
2 texture element is a uniform value associated with the uniform texture element.

1 29. The method of claim 28, wherein the uniform value indicates at least one of:

2 (a) an intensity; and

3 (b) a color.

1 30. The method of claim 26, wherein an element of the grid comprises a pointer  
2 indicating a nonuniform texture element refers to a quadrilateral of texels.

- 1 31. The method of claim 30, wherein the quadrilateral comprises two triangles in  
2 texture space mapped to two contiguous surface elements of the object.
- 1 32. The method of claim 26, wherein the step of blending the plurality of texture  
2 layers comprises performing at least one compositing operation.
- 1 33. A method of creating a blended texture for use in rendering a virtual object, the<sup>4</sup>  
2 method comprising the step of blending a plurality of texture layers comprising at least  
3 one member of each of at least two of:
- 4 (i) a scratch texture;
  - 5 (ii) a stencil texture; and
  - 6 (iii) a paint texture.
- 1 34. The method of claim 33, wherein at least one of the plurality of texture layers has  
2 associated therewith a grid of pointers indicating at least one member of each of at least  
3 two of:
- 4 (a) a uniform texture element;
  - 5 (b) a nonuniform texture element; and
  - 6 (c) a location of a nearest free texture element in the grid.
- 1 35. The method of claim 33, wherein the plurality of texture layers comprise at least  
2 one member of each of (i), (ii), and (iii).
- 1 36. The method of claim 33, wherein the step of blending the plurality of texture  
2 layers comprises performing at least one compositing operation.
- 1 37. The method of claim 33, wherein the step of blending a plurality of texture layers  
2 comprises blending a first set of at least two texture layers each corresponding to a first

3 texture region, and blending a second set of at least two texture layers each corresponding  
4 to a second texture region.

1 38. The method of claim 37, wherein the step of blending a plurality of texture layers  
2 comprises, for each of a plurality of texture regions, blending a set of at least two texture  
3 layers corresponding to a respective texture region.

1 39. The method of claim 33, wherein the plurality of texture layers comprises a  
2 scratch texture corresponding to a single brush stroke performed by a user.

1 40. The method of claim 39, wherein the single brush stroke is selected from the  
2 group consisting of a paint stroke, an erase stroke, a pencil stroke, a pen stroke, a line  
3 application, a character application, and a text application.

1 41. The method of claim 39, wherein the single brush stroke extends from a first  
2 texture region into a second texture region.

1 42. The method of claim 39, wherein the brush stroke is divided into a plurality of  
2 segments.

1 43. The method of claim 42, wherein an intensity value corresponding to a location  
2 within the brush stroke is written at a texel of the scratch texture only if the intensity  
3 value exceeds an existing value at the texel.

1 44. The method of claim 39, wherein a value written at a texel of the scratch texture is  
2 a function of a distance between a point in object space corresponding to the texel and a  
3 point in object space along a center of the brush stroke.

1 45. The method of claim 44, wherein the point in object space corresponding to the  
2 texel and the point in object space along the center of the brush stroke are represented  
3 with Cartesian coordinates.

1 46. The method of claim 33, wherein the step of blending comprises blending a  
2 scratch texture corresponding to a single brush stroke with at least a second texture  
3 substantially upon completion of the brush stroke.

1 47. The method of claim 33, wherein the plurality of texture layers comprises a  
2 plurality of scratch textures, each corresponding to a single brush stroke performed by a  
3 user.

1 48. The method of claim 47, wherein the step of blending comprises blending the  
2 plurality of scratch textures in an order in which the brush strokes corresponding to the  
3 scratch textures are performed.

1 49. The method of claim 33, wherein the step of blending proceeds while a user  
2 performs brush strokes.

1 50. The method of claim 49, wherein the step of blending proceeds in a plurality of  
2 discrete time slices.

1 51. The method of claim 33, wherein the plurality of texture layers corresponds to a  
2 first texture region, and wherein the method further comprises the steps of:

3 (i) creating a first stencil texture comprising a plurality of texels, each with an  
4 assigned scalar value indicating a level of protection for a corresponding texel in at least  
5 a portion of the first texture region;

6 (ii) directing graphical input corresponding to a plurality of brush strokes  
7 performed by a user into a second stencil texture; and

8 (iii) modifying a value of at least one texel of the second stencil texture using  
9 the first stencil texture.

1 52. A method of interactively representing application by a user of at least one brush  
2 stroke directly onto a virtual object in object space, the method comprising the steps of:

3 (a) allocating a plurality of texture elements in two dimensional texture space  
4 for a plurality of jacks of a virtual object;

5 (b) graphically rendering the allocated texture in real time as a user applies at  
6 least one brush stroke onto the virtual object as represented in object space, wherein the  
7 rendering step comprises creating at least one blended texture that is at least temporarily  
8 bound to a graphics application during the rendering; and

9 (c) updating at least one of the blended textures according to the at least one  
10 brush stroke applied by the user, wherein the method further comprises at least one of:

11 (i) using an index corresponding to each of a plurality of jacks of the  
12 virtual object to identify texture elements to which surface elements of the virtual object  
13 are mapped;

14 (ii) allocating texture in the texture space for at least one newly-  
15 created jack of the virtual object following an object modification; and

16 (iii) blending a set of texture layers corresponding to a first of a  
17 plurality of texture regions in the texture space and binding the blended texture to  
18 the graphics application during rendering of the first texture region.

1 53. The method of claim 52, wherein the method further comprises at least two of:

2 (i) using an index corresponding to each of a plurality of jacks of the  
3 virtual object to identify texture elements to which surface elements of the virtual object  
4 are mapped;



- 5                   (ii)     allocating texture in the texture space for at least one newly-  
6     created jack of the virtual object following an object modification; and  
7                   (iii)    blending a set of texture layers corresponding to a first of a  
8     plurality of texture regions in the texture space and binding the blended texture to  
9     the graphics application during rendering of the first texture region.

1   54.    The method of claim 52, wherein the method further comprises:

- 2                   (i)     using an index corresponding to each of a plurality of jacks of the  
3     virtual object to identify texture elements to which surface elements of the virtual object  
4     are mapped;

- 5                   (ii)    allocating texture in the texture space for at least one newly-  
6     created jack of the virtual object following an object modification; and

- 7                   (iii)    blending a set of texture layers corresponding to a first of a  
8     plurality of texture regions in the texture space and binding the blended texture to  
9     the graphics application during rendering of the first texture region.

1   55.    The method of claim 52, wherein the at least one brush stroke comprises at least  
2     one member selected from the group consisting of a paint stroke, an erase stroke, a pencil  
3     stroke, a pen stroke, a line application, a character application, and a text application.

1   56.    The method of claim 52, further comprising the step of:

- 2                   (d)     haptically rendering the virtual object in real time as the user applies the at  
3     least one brush stroke onto the virtual object.

1   57.    The method of claim 56, further comprising the step of:

- 2                   (e)     determining force corresponding to a position of a haptic interface device  
3     held by the user as the user applies the at least one brush stroke onto the virtual object.

- 1 58. The method of claim 57, further comprising the step of:  
2 (f) providing the force to the user through the haptic interface device as the  
3 user applies the at least one brush stroke onto the virtual object.
- 1 59. The method of claim 56, wherein step (d) is performed at a substantially faster  
2 rate than step (b).
- 1 60. The method of claim 56, wherein step (d) is performed within a range from about  
2 700 Hz to about 1500 Hz and wherein step (b) is performed within a range from about 5  
3 Hz to about 150 Hz.
- 1 61. The method of claim 60, wherein step (d) is performed at about 1000Hz and  
2 wherein step (b) is performed at up to about 40 Hz.
- 1 62. The method of claim 56, wherein step (b) and step (d) are performed by different  
2 threads.